1. An apparatus in a microprocessor for detecting that the microprocessor erroneously branched to a speculative target address that is provided by a branch target address cache (BTAC), the apparatus comprising:

a storage element, for storing an indication of whether the microprocessor branched to the speculative target address provided by the BTAC without knowing whether an instruction associated with said indication is a branch instruction;

instruction decode logic, configured to receive and decode said instruction subsequent to the microprocessor branching to the speculative target address; and

prediction check logic, coupled to said instruction decode logic, for notifying branch control logic that the microprocessor erroneously branched to the speculative target address if said instruction decode logic indicates said instruction is not a branch instruction and said indication indicates the microprocessor branched to the speculative target address.

2. The apparatus of claim 1, wherein said storage element is in an instruction buffer for storing instructions, including said instruction.

3. The apparatus of claim 1, wherein said indication indicates that the microprocessor branched to the speculative target address cached in the BTAC without certainty that said instruction decoded by said instruction decode logic is a same instruction for which the BTAC cached the speculative target address.

4. The apparatus of claim 1, wherein said indication indicates that the microprocessor branched to the speculative target address provided by the BTAC in response to a fetch address that selected a line of instructions in an instruction cache.

5. The apparatus of claim 4, wherein said indication indicates that the microprocessor branched to the speculative target address in response to said fetch address without certainty whether a previously executed instruction, for which the BTAC cached said target address, is present in said instruction cache line.

6. The apparatus of claim 1, wherein said instruction decode logic is configured to determine a first instruction length of said instruction.

7. The apparatus of claim 6, wherein said prediction check logic is configured to notify said branch control logic that the microprocessor erroneously branched to the speculative target address if said first instruction length does not match a second instruction length that is cached in the BTAC and received therefrom.

8. The apparatus of claim 1, wherein said prediction check logic is configured to notify said branch control logic that the microprocessor erroneously branched to the speculative target address if said indication is associated with a byte of said instruction not defined as a valid opcode byte by an instruction set of the microprocessor.

9. The apparatus of claim 8, wherein said microprocessor instruction set is an x86 architecture instruction set.

10. The apparatus of claim 1, wherein the apparatus further comprises:

address generation logic, coupled to said instruction decode logic, for generating a correct target address of said instruction; and

a comparator, coupled to said address generation logic, for comparing the speculative target address provided by the BTAC and said correct target address of said instruction, and for providing a mismatch indicator to said prediction check logic based on said comparing.

11. The apparatus of claim 10, wherein said prediction check logic is configured to notify said branch control logic that the microprocessor erroneously branched to the speculative target address if said mismatch indicator indicates the speculative target address and said correct target address of said instruction do not match.

12. The apparatus of claim 1, wherein the apparatus further comprises:

execution logic, operatively coupled to said instruction decode logic, for determining a correct direction of said instruction, said correct direction specifying whether said

instruction is taken or not taken, said execution logic providing said correct direction to said prediction check logic.

13. The apparatus of claim 12, wherein said prediction check logic is configured to notify said branch control logic that the microprocessor erroneously branched to the speculative target address if said correct direction indicates said instruction is not taken.

14. An apparatus in a microprocessor for detecting that the microprocessor erroneously speculatively branched to a target address that is provided by a speculative branch target address cache (BTAC), the apparatus comprising:

a storage element, for storing an indication of whether the microprocessor speculatively branched to the target address provided by the BTAC based on an instruction cache fetch address without first determining whether a branch instruction is present in a line of instruction bytes in the instruction cache selected by said fetch address;

instruction decode logic, configured to receive and decode said instruction bytes in said instruction cache line subsequent to the microprocessor speculatively branching to the target address, said instruction decode logic further configured to indicate whether said line includes a branch instruction; and

prediction check logic, coupled to said instruction decode logic, for providing an error signal to branch control logic if said indication indicates the microprocessor speculatively branched to the

target address and said instruction decode logic indicates said line does not include a branch instruction.

15. The apparatus of claim 14, wherein the target address is provided by a speculative call/return stack in the microprocessor rather than the speculative BTAC in response to an indication cached in the BTAC that said line of instruction bytes includes a return instruction.

16. A microprocessor for detecting and correcting an erroneous speculative branch, comprising:

an instruction cache, for providing a line of instruction bytes selected by a fetch address, said fetch address provided to said instruction cache on an address bus;

a speculative branch target address cache (BTAC), coupled to said address bus, for providing a speculative target address of a previously executed branch instruction in response to said fetch address whether or not said previously executed branch instruction is present in said line;

control logic, coupled to said BTAC, configured to control a multiplexer to select said speculative target address as said fetch address during a first period; and

prediction check logic, coupled to said BTAC, configured to detect that said control logic controlled said multiplexer to select said speculative target address erroneously;

wherein said control logic is further configured to control said multiplexer to select a correct address as said fetch address during a second period in response to said prediction check logic detecting said erroneous selection.

17. The microprocessor of claim 16, wherein said second period is subsequent to said first period.

18. The microprocessor of claim 16, further comprising:

instruction decode logic, configured to receive and decode said instruction bytes and to specify to said prediction check logic whether a branch instruction is present in said instruction bytes.

19. The microprocessor of claim 18, wherein said prediction check logic detecting that said control logic controlled said multiplexer to select said speculative target address erroneously comprises said prediction check logic determining that a branch instruction is not present in said instruction bytes.

20. The microprocessor of claim 16, further comprising:

branch target address generation logic, configured to receive said line of instruction bytes and to

generate an instruction pointer of an instruction comprised in said line of instruction bytes;

wherein said correct address comprises said instruction pointer of said instruction.

21. The microprocessor of claim 20, wherein said instruction is comprised in said line of instruction bytes at a location of said previously executed branch instruction in said line.

22. The microprocessor of claim 21, wherein said location is cached in said BTAC.

23. The microprocessor of claim 16, further comprising:

branch target address generation logic, configured to receive said line of instruction bytes and to generate a correct branch target address of a branch instruction comprised in said line of instruction bytes based on execution of said branch instruction comprised in said line of instruction bytes;

wherein said correct address comprises said correct branch target address.

24. The microprocessor of claim 23, wherein said prediction check logic detecting that said control logic controlled said multiplexer to select said speculative target address erroneously comprises said prediction check logic determining that said correct branch target address and said speculative target address do not match.

25. The microprocessor of claim 23, wherein said branch instruction is comprised in said line of instruction bytes at a location of said previously executed branch instruction in said line.

26. The microprocessor of claim 25, wherein said location is cached in said BTAC.

27. The microprocessor of claim 16, further comprising:

execution logic, configured to receive said line of instruction bytes and to generate a correct direction of a branch instruction comprised in said line of instruction bytes, said correct direction generated based on execution of said branch instruction comprised in said line of instruction bytes.

28. The microprocessor of claim 27, wherein said prediction check logic detecting that said control logic controlled said multiplexer to select said speculative target address erroneously comprises said prediction check logic determining that said correct direction indicates said branch instruction comprised in said line of instruction bytes is not taken.

29. The microprocessor of claim 27, wherein said branch instruction is comprised in said line of instruction bytes at a location of said previously executed branch instruction in said line.

30. The microprocessor of claim 29, wherein said location is cached in said BTAC.

31. The microprocessor of claim 16, further comprising:

branch target address generation logic, configured to receive said line of instruction bytes and to generate an instruction pointer of a next instruction after an instruction comprised in said line of instruction bytes at a location of said previously executed branch instruction in said line;

wherein said correct address comprises said instruction pointer of said next instruction after said instruction.

32. The microprocessor of claim 31, wherein said location is cached in said BTAC.

33. The microprocessor of claim 16, further comprising:

instruction decode logic, configured to receive and decode said line of instruction bytes and to specify a length of an instruction comprised in said line of instruction bytes, said instruction being at a location of said previously executed branch instruction in said line.

34. The microprocessor of claim 33, wherein said prediction check logic detecting that said control logic controlled said multiplexer to select said speculative target address erroneously comprises said prediction check logic determining that said length of said instruction does not match an instruction length cached in said speculative BTAC for said previously executed branch instruction.

35. The microprocessor of claim 34, further comprising:

branch target address generation logic, configured to receive said instruction and to generate an instruction pointer of said instruction;

wherein said correct address comprises said instruction pointer of said instruction.

36. The microprocessor of claim 16, further comprising:

instruction decode logic, configured to receive and decode said instruction and to specify which of a plurality of bytes comprising said instruction is an opcode byte.

37. The microprocessor of claim 36, wherein said prediction check logic detecting that said control logic controlled said multiplexer to select said speculative target address erroneously comprises said prediction check logic determining that said control logic controlled said multiplexer to select said speculative target address based on a byte of said instruction other than said opcode byte specified by said instruction decode logic.

38. The microprocessor of claim 16, further comprising:

branch target address generation logic, configured to receive said line of instruction bytes and to generate an instruction pointer of an instruction comprised in said line of instruction bytes, said instruction located at a location of said previously executed branch instruction in said line;

wherein said correct address comprises said instruction pointer of said instruction.

39. The microprocessor of claim 16, wherein an entry of said speculative BTAC caching said speculative target address is invalidated in response to said prediction check logic detecting said erroneous selection.

40. The microprocessor of claim 16, wherein said speculative BTAC is updated with a direction prediction associated with said previously executed branch instruction, said speculative BTAC being updated in response to said prediction check logic detecting that said control logic controlled said multiplexer to select said speculative target address erroneously.

41. The microprocessor of claim 16, wherein said speculative target address is updated in said speculative BTAC in response to said prediction check logic detecting that said control logic controlled said multiplexer to select said speculative target address erroneously.

42. The microprocessor of claim 16, wherein said prediction check logic comprises an error output, coupled to said control logic, for notifying said control logic of said prediction check logic detecting that said control logic controlled said multiplexer to select said speculative target address erroneously.

43. The microprocessor of claim 16, wherein a plurality of pipeline stages of the microprocessor are flushed in response to said prediction check logic detecting that said control logic controlled said multiplexer to select said speculative target address erroneously.

44. The microprocessor of claim 16, further comprising:

an instruction buffer, coupled to said instruction cache, for buffering said line of instruction bytes;

wherein said instruction buffer is flushed in response to said prediction check logic detecting that said control logic controlled said multiplexer to select said speculative target address erroneously.

45. The microprocessor of claim 16, wherein said speculative BTAC and said instruction cache are accessed substantially in parallel.

46. A method for recovering from an erroneous branch to a speculative target address, the method comprising:

generating a speculative target address for a branch instruction that is presumed present in an instruction cache line selected by a fetch address;

branching to said speculative target address whether or not said presumed branch instruction is present in said instruction cache line;

generating a correct target address of the presumed branch instruction subsequent to said generating said speculative target address;

determining if said speculative target address matches said correct target address; and

branching to said correct target address if said speculative target address does not match said correct target address.

47. The method of claim 46, further comprising:

storing an indication of whether said branching to said speculative target address occurred, prior to said determining; and

said branching to said correct target address only if said indication indicates said branching to said speculative target address occurred.

48. The method of claim 46, wherein said generating said correct target address comprises calculating said correct target address using instruction bytes of the presumed branch instruction.

49. The method of claim 46, further comprising:

updating an entry containing said speculative target address in a branch target address cache with said correct target address if said speculative target address does not match said correct target address.

50. A method for recovering from an erroneous branch to a speculative target address for a branch instruction, the branch instruction presumably present in a line of instructions, the line of instructions provided by an instruction cache in response to a fetch address, the speculative target address speculatively generated by a branch target address cache (BTAC) in response to the fetch address, the method comprising:

decoding the presumed branch instruction subsequent to the BTAC speculatively generating the speculative target address;

determining if the presumed branch instruction is a non-branch instruction in response to said decoding; and

branching to an instruction pointer of the presumed branch instruction if the presumed branch instruction is a non-branch instruction.

51. The method of claim 50, further comprising:

calculating said instruction pointer of the presumed branch instruction in response to said decoding.

52. The method of claim 50, further comprising:

invalidating an entry containing the speculative target address in the BTAC if the presumed branch instruction is a non-branch instruction.

53. The method of claim 52, wherein said invalidating is performed prior to said branching to said instruction pointer.

54. A method for recovering from an erroneous branch to a speculative target address, the speculative target address associated with a branch instruction that is presumably present in a cache line selected by a fetch address, the speculative target address provided by a branch target address cache (BTAC) in response to the fetch address, the method comprising:

decoding the presumed branch instruction subsequent to the BTAC providing the speculative target address;

determining a length of the presumed branch instruction; and

branching to an instruction pointer of the presumed branch instruction if said length of the presumed branch instruction does not match an instruction length speculatively provided by the branch target address cache.

55. The method of claim 54, further comprising:

invalidating an entry containing the speculative target address in the BTAC if said length of the presumed branch instruction does not match said

instruction length speculatively provided by the branch target address cache.

56. The method of claim 55, wherein said invalidating is performed prior to said branching to said instruction pointer.

57. A method for recovering from an erroneous branch to a speculative target address, the method comprising:

generating a speculative target address of a branch instruction that is presumed present in an instruction cache line selected by a fetch address;

generating a speculative direction prediction of the presumed branch instruction;

branching to said speculative target address whether or not said presumed branch instruction is present in said instruction cache line;

generating a correct direction of the presumed branch instruction subsequent to said generating said speculative direction prediction;

determining if said correct direction is not taken; and

branching to an instruction pointer of a next instruction after the presumed branch instruction if said correct direction is not taken.

58. The method of claim 57, further comprising:

updating said speculative direction prediction in a branch target address cache in response to said correct direction if said correct direction is not taken.

59. An apparatus in a microprocessor for detecting an erroneous branch to a speculative return address that is provided by a speculative call/return stack, the apparatus comprising:

a storage element, for storing an indication of whether the microprocessor branched to the speculative return address without knowing whether or not an instruction associated with said indication is a branch instruction;

instruction decode logic, configured to receive and decode said instruction subsequent to the microprocessor branching to the speculative return address; and

prediction check logic, coupled to said instruction decode logic, for notifying branch control logic that the microprocessor erroneously branched to the speculative return address if said instruction decode logic indicates that said instruction is not a branch instruction and said indication indicates that the microprocessor branched to the speculative return address.

60. A microprocessor for detecting and correcting an erroneous speculative branch, comprising:

an instruction cache, for providing a line of instruction bytes selected by a fetch address;

a speculative call/return stack, for providing a speculative return address of a previously executed branch instruction in response to said fetch address, said speculative return address provided whether or not said previously executed branch instruction is present in said line of instruction bytes;

control logic, coupled to said speculative call/return stack, configured to control a multiplexer to select said speculative return address to be said fetch address during a first period; and

prediction check logic, coupled to said control logic, configured to detect that said control logic controlled said multiplexer to select said speculative return address erroneously;

wherein said control logic is further configured to control said multiplexer to select a correct address to be said fetch address during a second

period, said control logic selecting said correct address in response to said prediction check logic detecting that said control logic controlled said multiplexer to select said speculative return address erroneously.

61. A method in a microprocessor for recovering from an erroneous branch to a speculative target address of a presumed branch instruction, the method comprising:

providing a speculative target address in response to an instruction cache fetch address;

producing an instruction cache line in response to said instruction cache fetch address;

decoding an instruction from said instruction cache line subsequent to said providing said speculative target address;

wherein said decoding is performed for a first time by the microprocessor for said instruction;

branching to said speculative target address prior to said decoding; and

branching to a correct target address of said instruction subsequent to said branching to said speculative target address in response to said decoding.

62. A method for recovering from an erroneous branch to a speculative target address, the method comprising:

providing a speculative target address for a branch instruction that is presumed present in an instruction cache line that is selected by a fetch address;

branching to said speculative target address whether or not said presumed branch instruction is present in said instruction cache line; and

correcting from an erroneous branch if said presumed branch instruction is not present in said instruction cache line.

63. A branch apparatus in a microprocessor for detecting when the microprocessor erroneously branches to a speculative target address, the speculative target address being provided by a branch target address cache (BTAC), the apparatus comprising:

a branch hit indicator, provided to indicate when the microprocessor branches to the speculative target address, said branch hit indicator provided whether or not an instruction associated with said branch hit indicator is a branch instruction;

instruction decode logic, configured to receive and decode said instruction and to specify whether said instruction is a branch instruction; and

prediction check logic, coupled to said instruction decode logic, for determining that the microprocessor erroneously branched to the speculative target address;

wherein the microprocessor erroneously branched to the speculative target address when said instruction decode logic specifies that said instruction is not a branch instruction, and said branch hit

indicator indicates that the microprocessor branched to the speculative target address.